

Unit 5: Structuring webpage using HTML

- HTML Table is an arrangement of data in rows and columns, or possibly in a more complex structure.
- Tables are widely used in communication, research, and data analysis.
- Tables are useful for various tasks such as presenting text information and numerical data.
- It can be used to compare two or more items in the tabular form layout.
- Tables are used to create databases.

Defining Tables in HTML:

- An HTML table is defined with the <table> tag.
- Each table row is defined with the <tr> tag.
- A table header is defined with the <th> tag.
- By default, table headings are bold and centered.
- A table data/cell is defined with the <td> tag.

`<tr>` - represents rows

`<td>` - used to create data cells

`<th>` - used to add table headings

`<caption>` - used to insert captions

`<thead>` - adds a separate header to the table

`<tbody>` - shows the main body of the table

`<tfoot>` - creates a separate footer for the table

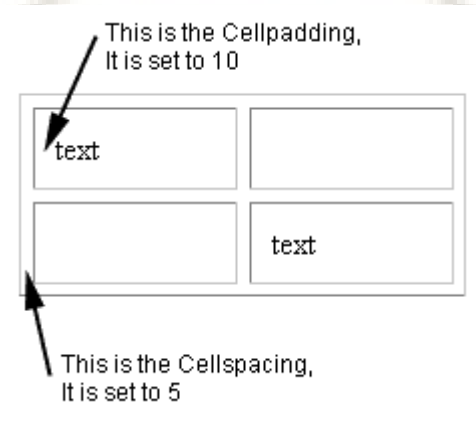
Attributes of <table>

Attribute	Value	Description
align	Left/ center/right	Defines how the table must be aligned according to the surrounding text.
bgcolor	colorname	Defines the background color for a table.
border	0 / 1/ 2	Defines the size of the frame surrounding the table. (Greater number means thicker border)
cellpadding	pixels	Defines the space between the cell wall and the cell content.

Attribute	Value	Description
cellspacing	pixels	Defines the space between cells.
width	pixels	Defines the width of a table.

We cannot mention height using “height” attribute. To give table height we use CSS property.

Difference between cell-spacing and cell-padding



<td> Attributes

Rowspan and colspan

- HTML tables can have cells that spans over multiple rows and/or columns.

NAME		

APRIL		

2022		
FIESTA		

- To make a cell span over multiple columns, use the colspan attribute.
- To make a cell span over multiple rows, use the rowspan attribute:

```

<!DOCTYPE html>
<html>
<head></head>
<body>
<table width="50%">
  <tr>
    <th colspan="2">Name</th>
    <th>Age</th>
    <th>Class</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>15</td>
    <td rowspan="2">FYBCA - 1 </td>
  </tr>
  <tr>
    <td>Eva</td>
    <td>Jackson</td>
    <td>13</td>
  </tr>
</table>
</body>
</html>

```

Name		Age	Class
Jill	Smith	15	FYBCA - 1
Eva	Jackson	13	

<table> captions

- The main use of adding a caption to table is to provide a description about the data represented in the table.
- The caption can either be placed at the top of the table or the bottom and by default it will always be centered.
- To insert a caption into a table, use the <caption> tag.

FYBCA Students			
Name		Age	Class
Jill	Smith	15	FYBCA - 1
Eva	Jackson	13	

- Here FYBCA Students is the caption of the table.

```

<!DOCTYPE html>
<html>
<head></head>
<body>
<table width="50%" border="1">
  <caption>FYBCA Students</caption>
  <tr>
    <th colspan="2">Name</th>
    <th>Age</th>
    <th>Class</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>15</td>
    <td rowspan="2">FYBCA - 1 </td>
  </tr>
  <tr>
    <td>Eva</td>
    <td>Jackson</td>
    <td>13</td>
  </tr>
</table>
</body></html>

```

Links and Bookmarks

- HTML links can be used to create bookmarks, so that readers can jump to specific parts of a web page.
- Bookmarks can be useful if a web page is very long.
- To create a bookmark - first create the bookmark, then add a link to it.
- Links can be added using <a> .
- When the link is clicked, the page will scroll down or up to the location with the bookmark.

Steps to create a bookmark internally on the same page:

1. First, use the id attribute to create a bookmark
 <h2 id="C4">Chapter 4</h2>
2. Then, add a link to the bookmark E.g.("Jump to Chapter 4"), from within the same page:
 Jump to Chapter 4

Note: You can also add a link to a bookmark on another page:

```
<a href="html_demo.html#C4">Jump to Chapter 4</a>
```

HTML Forms and CONTROLS

Web forms also called Fill-out Forms, let a user return information to a web server for some action. The processing of incoming data is handled by a script or program written in language that manipulates text files and information. Form element is used to delimit a data input form. There can be several forms in a document, but form elements can not be nested. The forms are not hard to code. These follow the same constructs as an other HTML elements.

Creating HTML Forms

As with most things in HTML, a structure starts with an opening tag, and ends with a closing tag. With forms, these tags are `<FORM>` and `</FORM>`.

```
<FORM      METHOD=method ACTION=url   NAME=frmname  
          ENCTYPE=enctype TARGET=target>
```

```
.....</FORM>
```

The url points to the remote file or application used for digesting the information. The method is either GET or POST. Inside the `<FORM>`, a Web page can contain any standard HTML formatting information, graphics, links to other pages and a bunch of new tags specific to forms.

`<FORM>` Element has the following attributes:

- **METHOD**- The method can be set to either POST or GET. The difference between the two has to do with the way the information is sent to the asp program. Always use the POST method.
- **ACTION**- The action attribute is URL specifying the location where the contents of the form are to be submitted to elicit a response. If the ACTION attribute is missing, the URL of the document itself is assumed. This tells the form, what program should be executed by the server when the form's data is submitted. The path to your asp file will vary from system to system.
- **NAME** attribute is used to refer to the form when working with controls in it from JavaScript. Despite their names, Forms are not visible objects on the screen, it is purely a logical concept to hold the controls together.
- **Target** specifies the name of the frame to put the results in.
- **ENCTYPE** This attribute specifies how the data is to be encoded. It sets the MIME type used to encode the name/value pair when sent to the action URL. This attribute is applied only if POST method is used. The default value is "application/X-www-form-urlencoded". For the File control it has the value "multipart/form-data"

These tags start and end a form (all input fields of the form are placed between these two tags). METHOD specifies which technical protocol the web server will use to pass the form data to the program which processes it (always set it to POST), and ACTION tells the server exactly which program that is. *Note:* POST must be capitalized, otherwise the method defaults to "GET".

GET sends the information entered in the form to the server at the end of the URL. Get is the default method, a question mark (?) and the form data is appended to the target URL of the code on the server and this complete variable is called query-string, which is environment variable of the server.

If there are two text fields named name and id on a form holding data RAM KUMAR and 100 respectively, then the data from these text fields would be appended to the URL like this

Filename.asp?name=RAM+KUMAR&id=100

But GET method has disadvantage from security point of view as it shows all the values entered by the user

POST This method is preferred over the GET method as it sends the contents of the form as a data block through the standard input stream using the http header.

Adding HTML controls to a Form

There are plenty of controls available in HTML. Many HTML controls are created with the <INPUT> element, using a different value for the TYPE attribute. The <INPUT> tag is all self-contained and has no closing tag, as is the case with the tag. There are several types of <INPUT> definitions.

Tag or Option	Meaning
<INPUT>	Text or other data-input field
TYPE	Type INPUT entry field
NAME	Symbolic name of field value
VALUE	Default content of text field
CHECKED	Button/Box checked by default
SIZE	Number of characters in text field
MAXLENGTH	Maximum characters accepted
<SELECT>	Grouped check boxes
<TEXTAREA>	Multiline Text Entry field

Using different value for the TYPE attribute of <INPUT> element various controls are created.

Creating Text Fields

Creates a text field that the user can enter or edit text in. It has the following attributes:

- NAME :attribute gives this element a name
- SIZE :sets the size.
- MAXLENGTH : Sets the maximum length of the data in the control, in characters.
- STYLE : Inline style indicating how to render the element
- READONLY : Indicates that the contents of the text field may not be modified.
- TITLE :Holds additional information, which might be displayed in tool tips for the element

- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- **TYPE** : Specifies the type of the element
- **VALUE** : sets the caption of the element. Set the alphanumeric characters.

`<INPUT TYPE="text" NAME="txtfield" MAXLENGTH=15 SIZE=15>`

Creating Button

Creates a customizable HTML Button in a form, which has the following attributes:

- **NAME** : attribute gives this element a name
- **STYLE** : Inline style indicating how to render the element
- **TITLE** : Holds additional information, which might be displayed in tool tips for the element
- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- **TYPE** : Specifies the type of the element
- **VALUE** : sets the caption of the element. Set the alphanumeric characters.

`<INPUT TYPE="button" NAME="button1" value="ClickMe">`

Creating CheckBox

Creates a checkbox in a form, which has the following attributes:

- **NAME** : Gives the element a name. Set to alphanumeric characters.
- **SIZE** : sets the size
- **STYLE** : Inline style indicating how to render the element.
- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- **CHECKED** : Indicates if the checkbox should appear checked initially or not.
- **TITLE** : Holds additional information, which might be displayed in tool tips for the element
- **TYPE** : Specifies the type of the element
- **VALUE** : sets the caption of the element. Set the alphanumeric characters.

Example

Family Pet

`<INPUT TYPE="checkbox" NAME="dog_owner" VALUE="owns_dog">Dog`

`<INPUT TYPE="checkbox" NAME="cat_owner" VALUE="owns_cat"`

`CHECKED>Cat`

`<INPUT TYPE="checkbox" NAME="bird_owner" VALUE="owns_bird">Bird`

`<INPUT TYPE="checkbox" NAME="cow_owner" VALUE="owns_cow">Cow`

Creating HIDDEN data

This is Hidden Control for "Hidden" input items. The 'hidden' input is not shown to the user. Instead Web designer will provide the value of the field. Usually, the hidden fields will contain configuration information for the server-side action script which handles the form data.

- NAME : Gives the element a name. Sets to alphanumeric characters.
- STYLE : Inline style indicating how to render the element.
- TITLE : Holds additional information, which might be displayed in tool tips for the element
- TYPE : Specifies the type of the element
- VALUE : sets the caption of the element. Set the alphanumeric characters.

Example:

```
<INPUT TYPE="HIDDEN" NAME="backup" VALUE="Hello! From HTML!">
```

Creating Password control

Creates a password text field, which masks typed input.

- NAME : attribute gives this element a name
- SIZE : sets the size.
- MAXLENGTH : Sets the maximum length of the data in the control, in characters.
- STYLE : Inline style indicating how to render the element
- TITLE : Holds additional information, which might be displayed in tool tips for the element
- TABINDEX : Set sthe tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- TYPE : Specifies the type of the element
- VALUE : sets the caption of the element. Set the alphanumeric characters.

Example

First Name

```
<INPUT TYPE="password" NAME="first_name" MAXLENGTH=15 SIZE=15>
```

Creating Radio Buttons

It creates a radio button in a form, which has the following attributes:

- NAME : attribute gives this element a name
- SIZE : sets the size.
- CHECKED: specifies selected element when first displayed
- STYLE : Inline style indicating how to render the element
- TITLE : Holds additional information, which might be displayed in tool tips for the element
- TABINDEX : Set sthe tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.

- TYPE :Specifies the type of the element
- VALUE :sets the caption of the element. Set the alphanumeric characters.

Example

```
Favorite <INPUT TYPE="radio" NAME="fav_radio" VALUE="WPLJ">WPLJ
Radio   <INPUT TYPE="radio" NAME="fav_radio" VALUE="WLIR">WLIR
Station <INPUT TYPE="radio" NAME="fav_radio" VALUE="WZRK">WZRK
        <INPUT TYPE="radio" NAME="fav_radio" VALUE="WBAB">WBAB
        <INPUT TYPE="radio" NAME="fav_radio" VALUE="WNEW">WNEW
        <INPUT TYPE="radio" NAME="fav_radio" VALUE="NONE">NONE
```

Creating Text Area

Creates a text area, much like a two dimensional text field where user can enter or edit text in. It has the following attributes:

- NAME :attribute gives this element a name
- STYLE : Inline style indicating how to render the element
- READONLY : Indicates that the contents of the text field may not be modified.
- TITLE :Holds additional information, which might be displayed in tool tips for the element
- TABINDEX : Set sthe tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- ROWS : Specifies the number of rows in the control. Set to positive integers.
- COLS : Specifies number of columns visible in the control. Set to a positive integer.
- WRAP :Determines the word wrap behaviour for the control. Set to OFF(no word wrapping, the default), **SOFT** and **HARD, VIRTUAL**

Example

```
Enter Your Comments
<TEXTAREA name="Question" wrap=virtual rows=2 cols=60>
</TEXTAREA>
```

Creating Select-Option Contol

It displays a select control, much like a grop-down list box. It has the following attributes:

- NAME :attribute gives this element a name
- SIZE :Gives the number of elements visible in the list. sets to appositive intger.
- MULTIPLE :Indicates that more than one option can be selected at a time.
- STYLE : Inline style indicating how to render the element
- TITLE :Holds additional information, which might be displayed in tool tips for the element

- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.

Option specifies a list item in the list of a **SELECT** control. Option has the following attributes:

- **SELECTED** : Indicates that the item should appear as selected at first.
- **STYLE** : Inline style indicating how to render the element
- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.
- **VALUE** : sets the caption of the element. Set the alphanumeric characters.

Example

```
<FORM METHOD="POST" ACTION="file1.asp">
<PRE>
  Salutation <SELECT NAME="salutation">
    <OPTION VALUE="">None
    <OPTION VALUE="Mr.">Mr.
    <OPTION VALUE="Mrs.">Mrs.
    <OPTION SELECTED VALUE="Ms.">Ms.
    <OPTION VALUE="Miss">Miss
  </SELECT>
```

Submitting Data from Forms

TYPE=SUBMIT

This type of **INPUT**, tells the browser that the information has been entered (maybe), and that the user wants to transmit the information to the server. It has the following attributes:

- **NAME** : attribute gives this element a name
- **SIZE** : Gives the size of the control
- **DISABLED** : Specifies that the element is disabled when first displayed
- **STYLE** : Inline style indicating how to render the element
- **TITLE** : Holds additional information, which might be displayed in tool tips for the element
- **TYPE** : Specifies the type of the element.
- **VALUE** : Gives the button another label besides the default, Submit Query. Set to alphanumeric characters.
- **TABINDEX** : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.

TYPE=IMAGE

Indicates an image users can click much as they would a Submit button. The mouse coordinates in the image are also passed to the form's action URL. It has the following attributes:

- NAME :attribute gives this element a name
- SIZE :Gives the size of the control
- SRC :Specifies the URL of the image.
- ALT : Sets the text that should be displayed for browsers that can not handle images. Sets to an alphanumeric strings.
- ALIGN : Sets the alignment of text following the image.
- DISABLED : Specifies that the element is disabled when first displayed
- STYLE : Inline style indicating how to render the element
- TITLE :Holds additional information, which might be displayed in tool tips for the element
- TYPE : Specifies the type of the element.
- VALUE : Indicates the symbolic result of the field when activated that is passed to the form processing script.
- USEMAP :Sets the URL of the client side map specification to be used.
- TABINDEX : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.

Example

```
<INPUT TYPE="IMAGE" SRC="submit.gif" NAME="Submit" VALUE="Submit">
```

TYPE=RESET

This INPUT tells the browser that the user wants to clear all of the fields and start over again.

- NAME :attribute gives this element a name
- SIZE :Gives the size of the control
- DISABLED : Specifies that the element is disabled when first displayed
- STYLE : Inline style indicating how to render the element
- TITLE :Holds additional information, which might be displayed in tool tips for the element
- TYPE : Specifies the type of the element.
- VALUE : Gives the button another label besides the default, Reset. Set to alphanumeric characters.
- TABINDEX : Set the tab index of the element, which locates it in the tab order of the form, allowing the user to press the Tab key and navigate from element to element. Set to positive or negative integers.

Example

```
<INPUT TYPE=reset VAUE="Reset this form">
```

Example 8.1

```
<HTML>
```

```
  <HEAD>
```

```
    <TITLE>
```

```
      WEB Page 8.1:form element Text and Button
```

```
    </TITLE>
```

```

</HEAD>
<BODY BGCOLOR="#ccffff">
<CENTER>
    <H1>
        Creating HTML Controls Text and Button
    </H1>
    <FORM NAME=forms1>
        <INPUT TYPE=TEXT Name="txtfld" SIZE=25
            TITLE="Enter some text">
        <BR>
        <BR>
        <INPUT TYPE=BUTTON Value="Click me"
            NAME="bt" TITLE="Click to continue...">
        <BR>
    </FORM>
</CENTER>
</BODY>
</HTML>

```

Example 8.2

```

<HTML>
<HEAD>
    <TITLE>
        WEB Page 8.2: Submitting data from forms
    </TITLE>
</HEAD>
<BODY BGCOLOR="#CCFFFF">
<CENTER>
<H1>
    Creating HTML Submit and Reset Controls
</H1>
<FORM NAME="forms1" METHOD="POST"
    ACTION="http://g6c7f5/nvv/thank.asp">
    Enter your name:
    <INPUT TYPE=TEXT Name="txtfld" SIZE=25 VALUE=""
        TITLE="Enter name please ...">
    <BR>
    <BR>
    <INPUT TYPE="submit" NAME="Submit" VALUE="Submit"
        TITLE="Finished with data entry...">
    <INPUT TYPE="reset" VALUE="Clear Form"
        TITLE="Clear all the entries...">
    <BR>
</FORM>
</CENTER>
</BODY>

```

</HTML>

Example 8.3

```
<HTML>
  <HEAD>
    <TITLE>
      WEB Page 8.3: form controls- TextArea, password and hidden
    </TITLE>
  </HEAD>
  <BODY BGCOLOR="#CCFFFF" TEXT=orange>
    <CENTER>
      <H1>
        Creating HTML Controls- TextArea Password and hidden
      </H1>
      <FORM NAME="forms1">
        Name : <INPUT TYPE=TEXT Name="txtfld" SIZE=25
              TITLE="Enter a text...">
        <BR>
        Password: <INPUT TYPE=PASSWORD NAME="psdfld" SIZE=20
                  TITLE="Enter Password Please...">
        <BR><BR><BR>
        <TEXTAREA NAME="txtarea" ROWS=5
              COLUMNS=30>
          Hello... You are in Text Area
        </TEXTAREA>
        <BR><BR>
        <INPUT TYPE=HIDDEN NAME="hdfld">
        <INPUT TYPE=BUTTON Value="Click me"
              NAME="bt1" TITLE="Click me to continue...">
        <BR>
      </FORM>
    </CENTER>
  </BODY>
</HTML>
```

Note: It is very difficult to set the form layout with all its controls at a proper position. Use of <TABLE> to position all the form controls is highly recommended.

Example 8.3 (Revised using table)

```
<HTML>
<HEAD>
  <TITLE>
    WEB Page 8.3: form controls- TextArea, password and hidden
  </TITLE>
</HEAD>
```



```
<BODY BGCOLOR="#CCFFFF" TEXT=orange>
<FORM NAME="forms1">
<TABLE border="0" ALIGN=CENTER>
  <TR>
    <TD colspan="2">
      <B>
        <FONT SIZE="6" COLOR="#000F85">
          <CENTER> Creating HTML Controls- TextArea Password and hiddeen
          <BR>
        </CENTER>
      </FONT>
      <B>
      <BR>
    </TD>
  </TR>
  <TR>
    <TD><FONT SIZE="4">Name</FONT></TD>
    <TD><INPUT TYPE=TEXT Name="txtfld" SIZE=25
      TITLE="Enter a text...">
    </TD>
  </TR>
  <TR>
    <TD><FONT SIZE="4">Password</FONT></TD>
    <TD><INPUT TYPE=PASSWORD NAME="psdfld" SIZE=20
      TITLE="Enter Password Please...">
    </TD>
  </TR>
  <TR>
    <TD><FONT SIZE="4">Comments</FONT></TD>
    <TD ALIGN=LEFT><TEXTAREA NAME="txtarea" ROWS=10
      COLS=40 TITLE="Enter Your comments
      Please ...">Hello... You are in Text Area
    </TEXTAREA>
  </TD>
  </TR>
  <TR ALIGN=CENTER>
    <TD COLSPAN=2 ><INPUT TYPE=BUTTON Value="Click me"
      NAME="bt1" TITLE="Click me to continue...">
    </TD>
  </TR>
  <TR>
    <INPUT TYPE=HIDDEN NAME="hdfld">
  </TABLE>
</FORM>
</BODY>
</HTML>
```

Example 8.4

```
<HTML>
  <HEAD>
    <TITLE>
      WEB Page 8.4:form control :radio buttons, checkbox, list control
    </TITLE>
  </HEAD>
  <BODY BGCOLOR="#CCFFFF" TEXT=BLUE>
    <CENTER>
      <H1>
        Creating Radio buttons, Checkbox and list control
      </H1>
      <BR>
      <BR>
      <FORM NAME="forms1" METHOD="POST"
        ACTION="http://g6c7f5/nvv/thank.asp">

        <H3>
          Family Pet
          <INPUT TYPE="checkbox" NAME="dog_owner" VALUE="owns_dog">Dog
          <INPUT TYPE="checkbox" NAME="cat_owner" VALUE="owns_cat"
            CHECKED>Cat
          <INPUT TYPE="checkbox" NAME="bird_owner" VALUE="owns_bird">Bird
          <INPUT TYPE="checkbox" NAME="cow_owner" VALUE="owns_cow">Cow
          <BR>
          <BR>
          Favorite Radio Station <INPUT TYPE="radio" NAME="fav_radio" VALUE="Pbi">PBI
          <INPUT TYPE="radio" NAME="fav_radio" VALUE="CYL">Cylon
          <INPUT TYPE="radio" NAME="fav_radio" VALUE="BBC"
            CHECKED>BBC
          <INPUT TYPE="radio" NAME="fav_radio" VALUE="URDU">Urdu
          <INPUT TYPE="radio" NAME="fav_radio" VALUE="AV">Akash
          Vani
          <INPUT TYPE="radio" NAME="fav_radio" VALUE="NONE">NONE
          <BR>
          <BR>
          Salutation <SELECT NAME="salutation">
            <OPTION VALUE="">None
            <OPTION VALUE="Mr.">Mr.
            <OPTION VALUE="Mrs.">Mrs.
            <OPTION SELECTED VALUE="Ms.">Ms.
            <OPTION VALUE="Miss">Miss
          </SELECT>
          <BR><BR>
          <INPUT TYPE="submit" NAME="Submit" VALUE="Submit"
            TITLE="Finished with data entry...">
          <INPUT TYPE="reset" VALUE="Clear Form">
```

TITLE="Clear all the entries...">

```
</H3>
</FORM>
</CENTER>
</BODY>
</HTML>
```

Example 8.5

```
<HTML>
  <HEAD>
    <TITLE>
      WEB Page 8.5: form control: image submit button
    </TITLE>
  </HEAD>
  <BODY BGCOLOR="#ffffcc" TEXT="#00cc99">
    <CENTER>
      <H1>
        Creating HTML Controls: mage submit button
      <FORM NAME="forms1" METHOD="POST"
        ACTION="http://g6c7f5/nvv/thank.asp">
        Enter your name:
          <INPUT TYPE=TEXT Name="txtfld" SIZE=25 VALUE=""
            TITLE="Enter name please ...">
          <BR>
          <BR>
          <INPUT TYPE="IMAGE" SRC="sub.gif" height=40
            width=60 NAME="Submit" VALUE="Submit"
            TITLE="Finished with data entry...">
          <BR>
        </FORM>
      </H1>
    </CENTER>
  </BODY>
</HTML>
```